What Is Claimed Is:

1. A method of synchronizing a plurality of simulation modules, comprising the steps of:

(a) issuing a clock credit to each simulation module;

(b) execution, by each simulation module, to an extent corresponding to the clock credit;

(c) for each simulation module, halting execution when the extent of execution corresponding to the clock credit has been completed; and

(d) when additional processing by at least one simulation module is necessary, issuing a further clock credit to each simulation module.

2. The method of claim 1, wherein step (a) comprises the step of issuing clock credit to each simulation module on the basis of synchronization points identified in data passing between the simulation modules.

3. The method of claim 1, further comprising, for one of the plurality of simulation modules, the steps of:

(e) creating a master clock signal;

(f) dividing the master clock signal to derive an additional clock signal; and

(g) applying the additional clock signal to one of the plurality of simulation modules,

wherein said steps (e) through (g) are performed after said step (a) and before said step (b).

4. The method of claim 3, wherein said steps (e), (f), and (g) are performed by a test bench component of the simulation module.

5. The method of claim 4, further comprising, before said step (e), the step of:

(h) creating a test bench component for the simulation module.

6. A method of creating a test bench component of a simulation module, the method comprising the steps of:

(a) reading specification information;

(b) identifying a device under test (DUT);

(c) determining the interface of the DUT; and

(d) generating the test bench component for the DUT, wherein the test bench component is capable of supporting testing of a plurality of clock domains of the DUT.

7. The method of claim 6, wherein said step (c) comprises the steps of:

(i) determining inputs and outputs of the DUT;

(ii) ascertaining attributes of the inputs and outputs; and

(iii) deriving a protocol of the inputs and outputs.

8. A system for synchronizing a plurality of simulation modules, comprising:

a clock arbitrator;

a programming language interface (PLI) for each simulation module, wherein said PLI receives a clock credit from said clock arbitrator and enables and halts simulation module execution on the basis of said clock credit; and

a test bench component for each simulation module, wherein said test bench component manages inputs and outputs to a device under test (DUT) within each simulation module.

9. The system of claim 8, wherein said clock arbitrator comprises a shared memory interface.

10. The system of claim 8, wherein said test bench component and said DUT are compiled together as a single binary executable module.

11. A computer program product comprising a computer usable medium having computer readable program code that enables a computer to synchronize a plurality of simulation modules, said computer readable program code comprising:

first computer readable program code logic for causing the computer to issue a clock credit to each simulation module;

second computer readable program code logic for causing the computer to execute each simulation module to an extent corresponding to the clock credit;

third computer readable program code logic for causing the computer to halt execution of a simulation module when the simulation module has completed execution to an extent corresponding to the clock credit; and

fourth computer readable program code logic for causing the computer to issue a further clock credit to each simulation module to perform additional execution by at least one simulation module.

12. The computer program product of claim 11, wherein said first computer readable program code logic comprises logic for causing the computer to issue clock credit to each simulation module on the basis of synchronization points identified in data passing between the simulation modules.

13. The computer program product of claim 11, further comprising:

fifth computer readable program code logic for causing the computer to create a master clock signal;

sixth computer readable program code logic for causing the computer to divide the master clock signal to derive an additional clock signal; and

seventh computer readable program code logic for causing the computer to apply the additional clock signal to one of the plurality of simulation modules.

14. A computer program product comprising a computer usable medium having computer readable program code that executes on a computer that creates a test bench component of a simulation module, said computer readable program code comprising:

first computer readable program code logic for causing the computer to read specification information;

second computer readable program code logic for causing the computer to identify a device under test (DUT);

third computer readable program code logic for causing the computer to determine the interface of the DUT; and

fourth computer readable program code logic for causing the computer to generate the test bench component for the DUT, wherein the test bench component is capable of supporting testing of a plurality of clock domains of the DUT.

15. The computer program product of claim 14, wherein said third computer readable program code logic comprises:

(i) computer readable program code logic for determining inputs and outputs of the module to be tested;

(ii) computer readable program code logic for ascertaining required attributes of the inputs and outputs; and

(iii) computer readable program code logic for deriving a protocol of the inputs and outputs.